

BAB II. TINJAUAN PUSTAKA

2.1 IoT

IoT merujuk pada konsep komputasi di mana setiap objek fisik dapat terhubung ke internet dan saling berinteraksi dengan perangkat lainnya untuk mengidentifikasi dan bertukar informasi (Mufaqih, Kaburuan, and Wang 2020). Keunggulan yang dapat diukur dari komputasi awan memungkinkan pengembang untuk menciptakan dan menyediakan layanan aplikasi mereka di dalamnya. *Cloud* berperan sebagai partner ideal bagi IoT dengan fungsi sebagai tempat penyimpanan dan akses data sensor jarak jauh. Kombinasi kedua teknologi ini mendorong terbentuknya *Cloud of Things* (CoT). Di mana perangkat (*node*) dapat diakses, dipantau, dan dikendalikan dari jarak jauh melalui *cloud*. Dengan skalabilitas tinggi yang dimiliki *cloud*, sejumlah *node* dapat ditambahkan atau dihapus dari sistem IoT secara *real-time* (Vijayabaskar et al. 2023).

IoT adalah konsep yang memungkinkan mesin untuk berkomunikasi secara langsung dengan mesin lainnya, dikenal juga sebagai komunikasi *machine to machine* (M2M) (Elakya et al. 2019). IoT menjadi fokus utama berkat pengembangan teknologi dan revolusi industri 4.0. Penerapan aplikasi IoT telah luas di berbagai sektor termasuk sistem keamanan, pemantauan, industri, dan pertanian (Ralenza Pratama et al. 2022). IoT bertujuan untuk memanfaatkan konektivitas internet yang terus-menerus, memungkinkan penghubung mesin, peralatan, dan benda fisik lainnya dengan sensor dan aktuator untuk mengumpulkan data dan mengelola operasinya sendiri. Dengan demikian, mesin dapat berkolaborasi dan bahkan bertindak berdasarkan informasi baru yang diperoleh secara mandiri (F. S. Apriano 2023).

Keunggulan utama dari IoT dalam *smart parking* adalah kemampuannya untuk memberikan informasi secara *real-time* tentang ketersediaan tempat parkir. Hal ini dapat mengurangi waktu yang dihabiskan oleh pengguna dalam mencari tempat parkir, serta mengurangi kemacetan di sekitar area parkir. Dengan demikian, penerapan IoT dalam sistem parkir tidak hanya memberikan manfaat langsung dalam hal efisiensi dan kenyamanan, tetapi juga memberikan kesempatan untuk meningkatkan perencanaan dan pengelolaan parkir secara holistik untuk masa depan yang lebih baik (Pradana and Sutabri 2024). Oleh karena itu, model IoT sangat fleksibel dapat dipercaya efektif dan sederhana untuk digunakan dan ditangani dengan cara pemanfaatan yang mendalam dalam berbagai aplikasi. Dengan bantuan informasi IoT dapat dikumpulkan dengan lebih efektif, pengoperasian sistem dapat diselesaikan dengan sangat tepat dengan konektivitas (Veeramanickam et al. 2022).

Banyak negara sekarang menerapkan *smart parking* yang didasarkan pada IoT dan *cloud* komputasi. Beberapa negara memanfaatkan berbagai strategi untuk mengelola lalu lintas dengan meningkatkan keselamatan berkendara. Sistem *smart parking* dapat memberikan pengemudi informasi kuantitatif melalui platform yang dapat mereka akses (Rathod 2021).

2.2 Mikrokontroler

Mikrokontroler adalah suatu alat elektronika digital yang dapat menerima masukan dan keluaran serta mengendalikan dengan program yang dapat ditulis dan dihapus dengan cara khusus. Mikrokontroler terdiri dari inti *processor*, perlengkapan input dan output, dan memori. Dalam praktiknya, mikrokontroler melakukan proses membaca dan menulis data (Tantowi and Kurnia 2020). Mikrokontroler menggabungkan memori untuk menyimpan data P / I / O atau data untuk melacak perangkat eksternal. Penggunaan mikrokontroler sekarang sangat populer di dunia bisnis elektronik. Ini terjadi karena mikrokontroler memiliki kelebihan dan kemudahan penggunaan. Selain itu, harganya lebih murah dibandingkan sistem lainnya (Bahri and Durbin Hutagalung 2023).

Tujuan mikrokontroler adalah untuk memproses sinyal dengan cara yang sesuai dengan model yang berfungsi. Pemanfaatan memori dan penerima serta pengirim sinyal informasi adalah bagian lain dari tugas kerja ini (Veeramanickam et al. 2022). Mikrokontroler berperan penting dalam menghubungkan perangkat keras dengan perangkat lunak untuk memastikan sistem parkir berfungsi dengan efisien dan otomatis (Ramsari and Utomo 2020).

2.2.1 ESP32-CAM

ESP32-CAM biasanya digunakan untuk proyek IoT yang membutuhkan fitur kamera karena memiliki fitur seperti Bluetooth, Wi-Fi, kamera, bahkan slot microSD (Marwita 2023). Modul kamera kecil ESP32-CAM sangat kompetitif dan dapat beroperasi secara mandiri dengan arus hingga 6mA. Sangat cocok digunakan untuk perangkat pintar, kontrol nirkabel industri, pemantauan nirkabel, identifikasi QR nirkabel, sinyal sistem penentu posisi nirkabel, dan aplikasi IoT lainnya. Dengan mengadopsi paket *Dual In-line Package* (DIP) dan dapat langsung dimasukkan ke dalam *backplane*, ESP32-CAM menawarkan mode koneksi keandalan tinggi yang nyaman untuk aplikasi di berbagai terminal perangkat keras IoT dan memungkinkan produksi produk yang cepat (Ngurah Yudistira, Kurniawan, and Subagyo 2022).



Gambar 2.1 ESP32-CAM

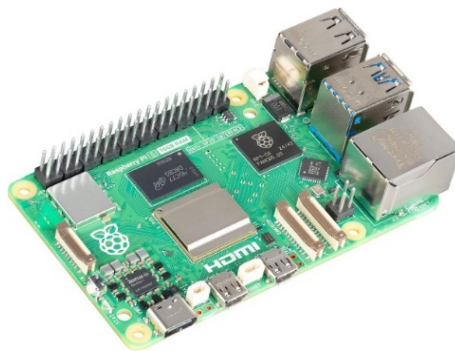
Sistem ini mampu melakukan deteksi objek secara *real-time*, seperti kendaraan dan pejalan kaki, serta membantu pengelolaan lalu lintas dan parkir secara efisien. Penggunaan ESP32-CAM sebagai perangkat utama menawarkan

solusi yang *cost-effective* dan dapat diterapkan di berbagai lingkungan perkotaan, terutama di daerah dengan anggaran terbatas dan mampu meningkatkan efisiensi transportasi serta mendukung keberlanjutan kota (Mohammad Annas et al. 2025).

2.2.2 Raspberry Pi

Dalam sistem *smart parking*, Raspberry Pi digunakan sebagai pusat kendali untuk mengelola data sensor, konektivitas ke *cloud* IoT, serta berfungsi sebagai server backend dalam memproses informasi lokasi dan status slot parkir (Sayeed et al. 2023). Raspberry Pi berfungsi sebagai pusat pengolahan data yang terhubung dengan berbagai sensor, seperti *Radio Frequency Identification* (RFID) dan sensor IR, serta perangkat lain seperti LCD dan motor servo. Raspberry Pi ini melakukan pengawasan terhadap slot parkir, memverifikasi identitas kendaraan melalui RFID dan mengirimkan data ke *database* melalui internet untuk mengelola data secara *real-time*. Selain itu, Raspberry Pi juga bertanggung jawab dalam pengendalian akses dan pengaturan gerbang otomatis, sehingga memungkinkan sistem parkir menjadi lebih efisien dan otomatis (Aravind et al. 2023).

Raspberry Pi digunakan dalam sistem pengenalan pelat nomor kendaraan sebagai platform utama untuk menjalankan model deteksi objek SSD MobileNet V2 serta proses OCR untuk identifikasi karakter. Dengan kemampuan komputasi yang memadai dan kompatibilitas yang tinggi, Raspberry Pi memungkinkan implementasi sistem otomatis parkir yang efisien dan biaya rendah cocok untuk digunakan di lingkungan yang memerlukan performa cukup baik namun dengan anggaran terbatas (Elfaki et al. 2023). Kombinasi kemampuan pemrosesan, kompatibilitas perangkat keras, dan kemudahan integrasi menjadikan Raspberry Pi pilihan utama dalam mengembangkan sistem smart parking yang efisien dan *scalable* (Pradhan et al. 2025).



Gambar 2. 2 Raspberry Pi 5

2.3 ALPR

ALPR adalah teknologi yang digunakan untuk secara otomatis mengenali dan membaca nomor pelat kendaraan dari citra atau video yang diambil melalui kamera.

Sistem ini biasanya terdiri dari beberapa komponen utama termasuk kamera, algoritma pengolahan citra, dan sistem OCR untuk mengekstrak teks dari pelat nomor yang terlihat. Teknologi ALPR terus berkembang dengan penggunaan teknik *Artificial Intelligence* (AI) dan *deep learning*, yang mampu meningkatkan akurasi bahkan pada kondisi pencahayaan dan cuaca yang buruk. Sistem ALPR modern sering digabungkan dengan IoT dan sistem pengolahan data secara *real-time* untuk mendukung konsep *smart city* dan pengelolaan sumber daya yang lebih efisien (Ditta et al. 2025).

Selain itu, pengintegrasian kamera berdefinisi tinggi dan teknologi *High Dynamic Range* (HDR) membantu meningkatkan akurasi pengenalan di lingkungan eksternal yang berubah-ubah. Penggunaan teknologi ALPR yang terintegrasi dengan sistem *smart parking*, memungkinkan identifikasi kendaraan secara otomatis saat masuk dan keluar area parkir (Pradhan et al. 2025). Model deteksi objek seperti YOLOv8 juga banyak digunakan, karena mampu mendeteksi dengan tingkat akurasi dan kecepatan tinggi (Taufiqurrahman, Hadi, and Siregar 2024). Lebih jauh lagi, sistem ini juga secara langsung mengurangi biaya tenaga kerja, mengurangi kemacetan, dan meningkatkan keamanan kendaraan, termasuk pengurangan risiko pencurian dan kecelakaan (Anany Srivastava et al. 2023).

2.4 Pengolahan Citra dengan OpenCV

Dalam sistem pengenalan pelat nomor berbasis citra, proses konversi gambar pelat nomor menjadi teks merupakan tahap kunci yang menentukan tingkat akurasi dan efektivitas sistem secara keseluruhan. Setelah tahap deteksi dan ekstraksi pelat selesai, citra pelat nomor yang bersih diproses menggunakan OCR untuk mengenali karakter-karakter di atasnya (Elfaki et al. 2023). Selain itu, kualitas citra yang dihasilkan setelah proses deteksi dan ekstraksi sangat berpengaruh terhadap keberhasilan OCR. Faktor-faktor seperti jarak pengambilan gambar, pencahayaan, serta sudut pengambilan gambar dapat mempengaruhi keakuratan pengenalan karakter, sehingga pengaturan posisi kamera dan pencahayaan yang baik sangat diperlukan untuk meningkatkan performa sistem (Ranjan Choudhary et al. 2023). Oleh karena itu, peningkatan hardware seperti penggunaan kamera resolusi tinggi dan prosesor yang lebih kuat menjadi faktor penting untuk memperbaiki performa OCR serta keseluruhan. Evaluasi yang menyeluruh terhadap performa sistem di berbagai kondisi nyata juga diperlukan guna menentukan algoritma dan metode *pre-processing* terbaik untuk digunakan. Secara umum, proses *pre-processing* ini memegang peranan krusial dalam memastikan bahwa karakter pada pelat nomor dapat dikenali secara akurat meskipun citra yang diambil dalam kondisi yang tidak ideal, sehingga sistem dapat berfungsi lebih handal dan efektif dalam aplikasi lapangan (Mohammad Annas et al. 2025).

Untuk mengatasi permasalahan tersebut, diperlukan penerapan beberapa teknik *pre-processing* guna meningkatkan kualitas citra sebelum tahap OCR dilakukan. Berikut beberapa tahapan yang dilakukan antara lain sebagai berikut:

1. Konversi ke *Grayscale*

Tahapan pertama adalah mengubah gambar berwarna menjadi gambar *grayscale*. Proses ini menyederhanakan data citra dengan mengurangi informasi warna sehingga hanya tersisa intensitas *luminance*. Konversi ini penting karena OCR dan proses segmentasi karakter biasanya bekerja lebih efektif pada citra dengan tingkat kontras yang tinggi dan variabel warna yang tidak diperlukan.

2. Peningkatan Kontras

Setelah gambar diubah menjadi *grayscale*, dilakukan peningkatan kontras untuk memperjelas perbedaan antara karakter pelat nomor dan latar belakang. Teknik ini membantu menonjolkan karakter sehingga lebih mudah dipisahkan dan dikenali oleh sistem. Dengan kontras yang lebih baik, langkah segmentasi dan OCR menjadi lebih akurat.

3. Penghapusan *Noise* (*Noise Reduction*)

Gambar hasil dari kamera bisa mengandung *noise* akibat faktor pencahayaan, kualitas kamera, atau gangguan lingkungan. Oleh karena itu, proses penghilangan *noise* dilakukan menggunakan filter seperti *median filter* atau *Gaussian filter*. Tujuan utama filter ini adalah memperhalus gambar dan mengurangi gangguan yang dapat mengganggu proses segmentasi karakter maupun OCR.

4. *Thresholding*

Setelah gambar bersih dan memiliki kontras yang cukup, dilakukan teknik *thresholding* untuk memisahkan karakter dari latar belakang. *Thresholding* mengubah gambar *grayscale* menjadi gambar *biner* (hitam-putih), di mana piksel yang memiliki nilai di atas batas tertentu akan diset menjadi putih, dan yang di bawahnya menjadi hitam. Teknik ini sangat krusial karena memperjelas batas-batas karakter, sehingga karakter dapat dipotong dan diidentifikasi dengan lebih akurat.

5. Segmentasi

Setelah tahap *thresholding*, dilakukan proses segmentasi untuk memisahkan area pelat nomor dari seluruh gambar kendaraan. Selanjutnya, setiap karakter dalam pelat nomor dipisahkan secara individual. Segmentasi ini memungkinkan OCR untuk mengenali karakter satu per satu secara lebih efektif.

Dengan demikian, rangkaian tahapan *pre-processing* tersebut berperan penting dalam menghasilkan citra pelat nomor yang siap untuk dikenali secara akurat oleh sistem OCR, sehingga proses pengenalan karakter dapat berjalan lebih efektif dan menghasilkan tingkat akurasi yang tinggi.

2.5 Sensor

Sensor adalah perangkat yang digunakan untuk mendeteksi perubahan besaran fisik seperti tekanan, gaya, besaran listrik, cahaya, gerakan, kelembaban, suhu, kecepatan, dan fenomena-fenomena lingkungan lainnya. Setelah mengamati terjadinya perubahan, input yang terdeteksi tersebut akan dikonversi menjadi output yang dapat dimengerti oleh manusia baik melalui perangkat sensor itu sendiri ataupun ditransmisikan secara elektronik melalui jaringan untuk ditampilkan atau diolah menjadi informasi yang bermanfaat bagi penggunaannya (Ardiyanto, Arman, and Supriyadi 2021).

Sensor yang digunakan pada penelitian ini adalah sensor IR serta kamera untuk mendeteksi pelat nomor kendaraan pengguna. Sistem ini mengandalkan sensor IR dalam mendeteksi status ketersediaan slot parkir sekaligus kendaraan yang memasuki area gerbang parkir, memungkinkan gerbang terbuka secara otomatis. Sensor IR bekerja berdasarkan deteksi pantulan sinyal cahaya inframerah dari objek seperti kendaraan atau batas slot, yang kemudian digunakan untuk menentukan apakah suatu slot sedang terisi atau kosong (Robidin et al. 2023).

2.5.1 Sensor IR

Sensor IR merupakan perangkat elektronik yang berfungsi untuk mendeteksi adanya sinyal inframerah dalam jangkauan tertentu. Dalam sistem kerjanya, sensor ini terdiri dari dua bagian utama yaitu pemancar (IR *Transmitter*) yang mengirimkan sinar inframerah, dan penerima (IR *Receiver*) yang menangkap pantulan sinar tersebut saat mengenai suatu objek. Ketika ada benda yang masuk ke area pantauan, sinar inframerah akan dipantulkan kembali ke penerima yang menandakan keberadaan objek tersebut. Sensor ini umumnya memiliki 3 pin, yaitu VCC sebagai sumber tegangan positif, GND untuk *ground* atau tegangan negatif, serta pin OUT yang mengeluarkan sinyal digital HIGH atau LOW tergantung pada ada tidaknya objek di dekat sensor (Iftitah, Bakri, and Karim 2022).



Gambar 2.3 Sensor IR

Sensor IR biasanya bekerja pada tegangan DC 5V atau 3.3V, dengan jarak deteksi efektif sekitar 20 cm. Sensor ini mampu mengenali perubahan radiasi inframerah di sekitarnya dan mengubahnya menjadi sinyal tegangan pada output nya. Karena kemampuannya dalam mendeteksi keberadaan atau pergerakan objek, sensor ini kerap dimanfaatkan dalam sistem pendeteksi gerakan maupun perangkat keamanan (Dhanabalraj et al. 2021).

Tabel 2.1 Spesifikasi dari Sensor IR

Description		Value
Field of view		18° half angle
Output	Target temp.	40 μ V per °C difference from sensor body
	Sensor body temp.	0-2500 mV
Accuracy	-10°C to 65°C	\pm 0.2°C absolute accuracy \pm 0.1°C uniformity \pm 0.05°C repeatability
	-40°C to 70°C	\pm 0.5°C absolute accuracy \pm 0.3°C uniformity \pm 0.1°C repeatability and uniformity
Optics		Germanium lens
Wavelength range		8-14 μ m (corresponds to atmospheric window)
Response time		< 1 second to changes in target temperature
Input power		2.5 V excitation
Operating environment		-55-80°C; 0-100% RH (non-condensing)
Cable		4.5 meters twisted, shielded 4 conductor wire with Santoprene casing.
Dimensions		6 cm long by 2.3 cm diameter

Mass		190 g
-------------	--	-------

(Sumber: www.researchgate.net)

2.5.2 Sensor Ultrasonik

Sensor ultrasonik merupakan jenis sensor yang memanfaatkan pantulan gelombang suara berfrekuensi tinggi untuk mendeteksi keberadaan suatu objek, seperti kendaraan pada sistem parkir. Sensor ini bekerja dengan mengonversi gelombang suara menjadi sinyal listrik, sehingga dapat menghitung jarak atau mendeteksi objek berdasarkan frekuensi yang dipancarkan dan diterima kembali. Cara kerjanya didasarkan pada pengiriman gelombang suara ultrasonik yang akan dipantulkan kembali saat mengenai suatu benda, lalu ditangkap kembali oleh sensor. Sensor ini umumnya memiliki 4 kabel yaitu VCC, TRIG, ECHO, dan GND yang masing-masing berfungsi untuk menyuplai daya, memicu sinyal, menerima pantulan gelombang, serta sebagai *ground* dan semuanya dihubungkan ke mikrokontroler seperti Arduino untuk memproses data keberadaan pada suatu titik parkir (Robidin et al. 2023).

Ketika sebuah kendaraan mendekati objek di sekitarnya, sensor akan secara otomatis mengaktifkan suara peringatan berupa bunyi bip. Semakin dekat jarak kendaraan dengan objek tersebut, bunyi peringatan akan semakin cepat dan memberi sinyal kepada pengemudi untuk berhenti. Namun, meskipun cukup andal sensor ultrasonik juga memiliki keterbatasan. Tidak semua permukaan dapat memantulkan gelombang suara dengan baik, terutama jika permukaannya tidak reflektif sehingga sensor bisa gagal mendeteksi objek tertentu. Selain itu, sensor ini hanya efektif untuk mendeteksi benda yang berada tepat di hadapannya. Objek yang berada di sisi lain atau di luar cakupan gelombang mungkin tidak terdeteksi, sehingga pengemudi tetap perlu memperhatikan kondisi sekitar menggunakan kaca spion atau visual langsung (Prasad Patro et al. 2020). Prinsip kerja ini didukung oleh penelitian yang menyatakan bahwa gelombang ultrasonik digunakan oleh radar untuk mendeteksi keberadaan objek dengan menghitung jarak berdasarkan refleksi gelombang suara (Supriyono and Marjuki 2020).



Gambar 2.4 Sensor Ultrasonik

Menurut penjelasan lebih lanjut, sensor ultrasonik memancarkan gelombang suara setiap 60 milidetik. Selisih waktu antara saat gelombang dikirim dan diterima kembali dimanfaatkan untuk menentukan keberadaan kendaraan di

area parkir. Kepala sensor secara berkala memancarkan gelombang tersebut dan menghitung waktu pantulannya (Irfan et al. 2022).

Tabel 2.2 Spesifikasi dari Sensor Ultrasonik

Electrial Parameters	HC-SR04 Ultrasonic Module
Operating Voltage	5VDC
Operating Current	15mA
Operating Frequency	40KHz
Max. Range	4m
Nearest Range	2cm
Measuring Angle	15 Degrees
Input Trigger Signal	10us min. TTL pulse
Output Echo Signal	TTL level signal, proportional to distance
Board Dimensions	1-13/16" X 13/16" X 5/8"
Board Connections	4 X 0.1" Pitch Right Angle Header Pins

(Sumber: cobaarduino.blogspot.com)

2.6 Aktuator

Aktuator berperan penting dalam mengotomatisasi proses pengoperasian fasilitas fisik seperti pintu gerbang parkir atau palang otomatis. Aktuator adalah perangkat yang mengubah sinyal listrik menjadi gerakan mekanis untuk mengendalikan komponen fisik tertentu, misalnya membuka atau menutup gerbang parkir (Aravind et al. 2023). Dalam sistem *smart parking* berbasis RFID yang dikembangkan, motor servo digunakan sebagai aktuator mekanis untuk menggerakkan *barrier gate* secara otomatis setelah proses verifikasi RFID. Motor servo ini diaktifkan oleh Arduino UNO yang menerima sinyal dari RFID *reader* dan memastikan pengoperasian *gate* secara *real-time* (Arpilleda 2025).

2.6.1 Motor Servo

Motor servo adalah jenis motor yang dirancang untuk memberikan kontrol posisi dan rotasi yang presisi. Pada umumnya, motor servo memiliki sistem *gear* internal yang memungkinkan penggerakannya dalam sudut tertentu biasanya antara 0° hingga 180° dengan ketelitian tinggi. Sistem ini dilengkapi dengan umpan balik (*feedback*) yang memungkinkan motor untuk mengetahui posisinya saat ini dan menyesuaikan gerakannya untuk mencapai posisi yang diinginkan

secara akurat (Abdulsahab et al. 2024). Motor servo dipilih karena kemampuannya untuk bergerak dengan tingkat presisi tinggi dan kemampuan untuk mengatur posisi sudut tertentu secara otomatis. Setelah kendaraan melewati *barrier* dan meninggalkan area parkir, motor servo akan mengembalikan posisi *barrier* ke posisi semula, menutup akses bagi kendaraan berikutnya. Sistem ini menyediakan solusi otomatisasi yang efisien, mengurangi kebutuhan intervensi manusia, serta meningkatkan keamanan dan kelancaran lalu lintas di area parkir (Allbadi, Shehab, and Jasim 2021).



Gambar 2.5 Motor Servo

2.7 Python

Python adalah bahasa pemrograman terbuka yang kuat dengan kemudahan penggunaan baris kode. Bahasa ini terbuka dengan komunitas yang dinamis, perpustakaan yang luas, sintaks yang sederhana, perangkat lunak yang paling mudah diakses, dan pemeliharaan serta *debugging* yang sangat mudah. Karena alasan yang disebutkan di atas, peneliti memilih Python sebagai kode sumber utama dan menggunakannya untuk memprogram Arduino. Salah satu cara untuk menggunakan Python dan Arduino adalah dengan menggunakan protokol firma, yang memungkinkan kendali Arduino untuk ditransfer sepenuhnya ke perangkat lunak eksternal. Mengembangkan program Arduino yang dikendalikan oleh *port serial* memungkinkan proses di atas dilakukan. Program Arduino yang dikembangkan firma memungkinkan komunikasi antara Arduino dan perangkat lunak eksternal. Setiap kali modifikasi dibuat, protokol perusahaan mempermudah pengunggahan sketsa Arduino mengunggah protokol ke web. Python juga mendukung penggunaan paradigma pemrograman lainnya seperti pemrograman berorientasi objek (OOP) atau prinsip pemrograman fungsional (Patel, Mehta, and Chauhan 2021).

Python juga cocok untuk membangun aplikasi web dari sisi backend menggunakan framework populer seperti Django atau Flask. Flask dipilih karena kesederhanaannya dan kemampuan untuk membuat layanan API REST yang dapat diintegrasikan dengan banyak aplikasi web atau *mobile* dengan hanya beberapa baris kode. Dalam penelitian ini, Flask digunakan untuk membangun server yang dapat mengirimkan notifikasi kepada pengguna berdasarkan data yang dikumpulkan oleh sistem manajemen (Mičko and Papcun 2023).

2.7.1 Flask

Flask dimanfaatkan sebagai framework untuk membangun antarmuka web dari sistem yang akan dikembangkan. Melalui Flask, baik pengguna biasa maupun admin dapat berinteraksi langsung dengan sistem melalui tampilan berbasis web. Hal ini memungkinkan mereka untuk mengakses, memantau, dan mengendalikan sistem secara langsung. Selain itu, Flask juga berperan dalam menyajikan data secara *real-time* yang diperoleh dari integrasi dengan OpenCV dan *Message Queuing Telemetry Transport* (MQTT), sehingga pengguna dapat memantau situasi secara visual melalui halaman web (Thi Ha et al., 2023).

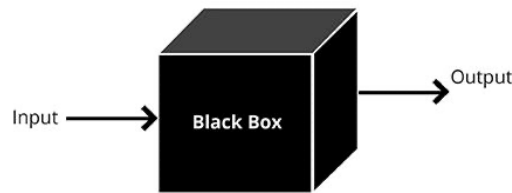
Flask digunakan sebagai framework untuk membangun layanan REST API yang berfungsi sebagai bagian dari sistem backend. Flask bertindak sebagai penghubung yang memungkinkan komunikasi antara data yang dikumpulkan oleh sistem pengenalan citra dan pengguna akhir yang membutuhkan informasi tersebut secara waktu nyata (Mičko & Papcun, 2023). Dalam sistem *smart parking*, Flask berfungsi sebagai server yang dikonfigurasi sebagai pendengar perubahan *database* dan mengirimkan notifikasi ke aplikasi pengguna (Patel, Mehta, and Chauhan 2021).

2.8 Testing

Testing adalah proses penting dalam pengembangan perangkat lunak untuk memastikan kualitas dan keandalan sistem. Dalam konteks penelitian ini, peneliti akan mencoba pendekatan *black-box* testing dan *white-box* testing. Pengujian *white-box* cenderung memberikan hasil yang lebih baik dalam hal cakupan kode dan deteksi kesalahan dibandingkan dengan pengujian *black-box*. Namun, pengujian *black-box* memiliki beberapa keuntungan praktis seperti kemampuan untuk dijalankan pada mesin *remote* tanpa perlu akses ke kode sumber atau file eksekusi, serta kemudahan penggunaan yang lebih cepat. Pengujian *black-box* juga memungkinkan pembuatan kasus uji dengan biaya manual yang hampir nol, meskipun hasilnya mungkin tidak sebaik pengujian *white-box*. Jadi, kedua pendekatan pengujian ini memiliki kelebihan dan kekurangan masing-masing (Arcuri 2021).

2.8.1 Black-box Testing

Pengujian perangkat lunak yang hanya berfokus pada spesifikasi fungsional tanpa memperhatikan struktur internal atau rancangan kode dikenal dengan istilah *black-box* testing. Jenis pengujian ini bertujuan untuk memastikan bahwa semua fungsi masukan dan keluaran perangkat lunak bekerja sesuai dengan spesifikasi yang telah ditentukan. Dalam pelaksanaannya, pengujian menyusun serangkaian skenario uji untuk mencoba setiap fitur yang tersedia, guna memastikan bahwa hasilnya sesuai dengan yang diharapkan. Metode ini sering disebut juga sebagai pengujian berbasis perilaku, pengujian input-output, atau pengujian fungsional, karena hanya menilai bagaimana sistem menanggapi data yang diberikan, tanpa harus memahami detail teknis atau logika internal dari sistem yang diuji (Ngurah Yudistira, Kurniawan, and Subagyo 2022).

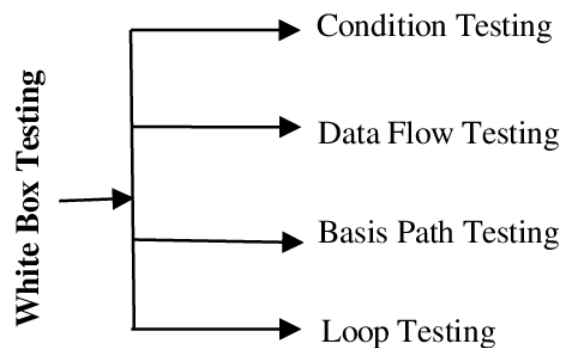


Gambar 2.6 Black-box Testing

Pengujian ini memiliki sejumlah keunggulan, antara lain biaya awal yang relatif rendah dan kemudahan dalam penerapannya. Selain itu, pengujian ini terbukti cukup efektif dalam menemukan berbagai kesalahan pada perangkat lunak. Namun demikian, terdapat pula keterbatasan seperti cakupan kode yang tidak terlalu luas dan kemungkinan tidak mampu mendeteksi semua *bug*, terutama yang berkaitan dengan jalur tertentu dalam program. Hal ini disebabkan karena pengujian dilakukan tanpa mempertimbangkan kondisi internal sistem, melainkan hanya berdasarkan input dan output (Arcuri, 2021).

2.8.2 White-box Testing

White-box testing memiliki sejumlah keunggulan, di antaranya kemampuan untuk mengidentifikasi dan menghilangkan bagian-bagian kode yang tidak diperlukan atau tersembunyi. Pengujian ini memungkinkan eksplorasi secara menyeluruh terhadap seluruh struktur dan logika program, sehingga cakupan pengujiannya lebih mendalam. Selain itu, pengujian ini juga bermanfaat dalam membantu proses optimasi kode agar lebih efisien. Menariknya, *white-box* testing sudah dapat dilakukan meskipun antarmuka pengguna grafis (GUI) dari sistem belum sepenuhnya selesai dikembangkan.



Gambar 2.7 *White-box* Testing

2.9 Peneliti Terdahulu

Tabel 2.3 Peneliti Terdahulu

Nama Peneliti	Judul Penelitian	Hasil Penelitian	Hasil Pembahasan
(Kusuma et al. 2023)	An Internet of Things – Based Touchless Parking System Using ESP32-CAM	ESP32-CAM berperan sebagai komponen utama dalam sistem parkir tanpa sentuh berbasis IoT. <i>Motor servo</i> difungsikan untuk menggerakkan portal parkir agar terbuka saat keberadaan pengemudi terdeteksi. Untuk mengatur pergerakan <i>motor servo</i> tersebut, sistem memanfaatkan sensor gerak (PIR) dan sensor ultrasonik yang bertugas mendeteksi kehadiran pengemudi maupun kendaraan secara otomatis.	Meningkatkan keandalan serta aspek keamanan pada sistem parkir tanpa sentuh, peneliti menyarankan integrasi metode QR-Code ke dalam sistem. Melalui pendekatan ini, pengemudi dapat mengakses sebuah halaman web melalui ponsel mereka untuk menghasilkan QR-Code yang bersifat unik setiap kali hendak memasuki area parkir. QR-Code tersebut kemudian akan dipindai oleh perangkat ESP32-CAM yang telah dipasang di gerbang parkir.
(Chougula 2020)	Automatic Smart Parking and Reservation System Using IoT	Penelitian ini menemukan bahwa penerapan sistem parkir dan reservasi pintar mampu meningkatkan efisiensi serta kenyamanan bagi pengguna. Meskipun demikian, studi sebelumnya belum secara spesifik membahas metode pembayaran yang	Dalam upaya mengembangkan sistem parkir dan reservasi pintar yang lebih modern, pemilihan metode pembayaran yang efisien dan inovatif menjadi hal yang krusial. Penggunaan Midtrans memungkinkan proses pembayaran dilakukan secara <i>real-time</i> , sehingga transaksi dapat berjalan dengan cepat dan efektif.

Nama Peneliti	Judul Penelitian	Hasil Penelitian	Hasil Pembahasan
		digunakan dalam sistem tersebut. Kondisi ini menunjukkan masih adanya kekurangan dalam pengembangan sistem secara menyeluruh, khususnya terkait integrasi metode pembayaran yang aman dan mudah digunakan.	
(Agarwal et al. 2021)	IoT Based Smart Parking System	Teknologi RFID digunakan dalam penelitian sebelumnya untuk mengidentifikasi pengguna dan kendaraan dalam sistem <i>smart parking</i> .	Sebagai pengembangan lanjutan, peneliti akan mengeksplorasi QR-Code sebagai metode identifikasi untuk meningkatkan efisiensi dan kenyamanan akses parkir. Integrasi ini diharapkan mempermudah dan mempercepat proses autentikasi pengguna.
(Koten et al. 2023)	Penerapan Internet of Things pada Smart Parking System untuk Kebutuhan Pengembangan Smart City	Pembayaran dilakukan melalui <i>payment gateway</i> menggunakan <i>e-money</i> dengan cara memindai kartu di mesin tiket. Mesin akan mengenali kartu, memotong saldo sesuai tarif dan durasi parkir lalu memungkinkan pengguna keluar setelah transaksi selesai.	Dalam penelitian ini, Midtrans dipilih sebagai <i>payment gateway</i> karena menawarkan kemudahan, keamanan, dan fleksibilitas metode pembayaran serta cocok untuk sistem parkir di mal dengan volume transaksi tinggi dan integrasi yang mudah.
(Mohammad Annas et al. 2025)	Advanced IoT-Integrated Parking Systems with Automated License Plate Recognition and Payment Management	Penelitian ini berhasil merancang dan menguji sistem <i>smart parking</i> berbasis IoT dengan integrasi teknologi ALPR	Hasil pengujian menunjukkan bahwa sistem <i>smart parking</i> berbasis IoT dengan ALPR mampu mengatasi tantangan seperti pencahayaan dan sudut pandang, berkat akurasi YOLOv6 dalam deteksi objek dan keandalan OCR

Nama Peneliti	Judul Penelitian	Hasil Penelitian	Hasil Pembahasan
		menggunakan algoritma YOLOv6 dan OCR Tesseract.	Tesseract dalam membaca pelat nomor.